IBM Python Assignment

1. Consider a list(list=[]) and perform insertion, print, remove, append, sort, pop, reverse.

Solution:

```
list = [1, 2, 4] #Creation of list

print(list)      #prints the list

list.insert(2,3) #Inserting interger(3) at position index(2)

print(list)

list.remove(1)   #Deleting the first element

print(list)

list.append(5)   #Inserting integer at the end of the list

print(list)

list.sort()      #Sorting the list

print(list)

list.pop(3)      #Pop the last element of the list
```

```
print(list)

list.reverse()   #Reversing the list

print(list)
```

Output:

[1, 2, 4]

[1, 2, 3, 4]

[2, 3, 4]

[2, 3, 4, 5]

[2, 3, 4, 5]

[2, 3, 4]

[4, 3, 2]

main.py

```
list = [1, 2, 4] #Creation of list

print(list)      #prints the list

list.insert(2,3) #Inserting interger(3) at position index(2)

print(list)

list.remove(1)   #Deleting the first element

print(list)

list.append(5)   #Inserting integer at the end of the list

print(list)

list.sort()      #Sorting the list

print(list)

list.pop(3)      #Pop the last element of the list

print(list)

list.reverse()   #Reversing the list

print(list)
```

```
[1, 2, 4]
[1, 2, 3, 4]
[2, 3, 4]
[2, 3, 4, 5]
[2, 3, 4, 5]
[2, 3, 4]
[4, 3, 2]
```

2. Write a calculator program in python.

Solution:

```
# This function adds two numbers
```

```
def add(x, y):
    return x + y

# This function subtracts two numbers
def subtract(x, y):
    return x - y

# This function multiplies two numbers
def multiply(x, y):
    return x * y

# This function divides two numbers
def divide(x, y):
    return x / y



print("Select operation.")
print("1.Add")
print("2.Subtract")
print("3.Multiply")
print("4.Divide")

while True:
    # Take input from the user
    choice = input("Enter choice(1/2/3/4): ")
```

```
    # Check if choice is one of the four options
    if choice in ('1', '2', '3', '4'):
        num1 = float(input("Enter first number: "))
        num2 = float(input("Enter second number: "))

        if choice == '1':
            print(num1, "+", num2, "=", add(num1, num2))

        elif choice == '2':
            print(num1, "-", num2, "=", subtract(num1, num2))

        elif choice == '3':
            print(num1, "*", num2, "=", multiply(num1, num2))

        elif choice == '4':
            print(num1, "/", num2, "=", divide(num1, num2))

        # Check if user wants another calculation
        # Break the while loop if answer is no
        calculation = input("Let's do next calculation? (yes/no): ")
        if calculation == "no":
          break
    else:
        print("Invalid Input")
```

Output:

Select operation.

1.Add

2.Subtract

3.Multiply

4.Divide

Enter choice(1/2/3/4): 1

Enter first number: 3

Enter second number: 2

3.0 + 2.0 = 5.0

Let's do next calculation? (yes/no): yes

Enter choice(1/2/3/4): 2

Enter first number: 3

Enter second number: 2

3.0 - 2.0 = 1.0

Let's do next calculation? (yes/no): yes

Enter choice(1/2/3/4): 3

Enter first number: 3

Enter second number: 2

3.0 * 2.0 = 6.0

Let's do next calculation? (yes/no): yes

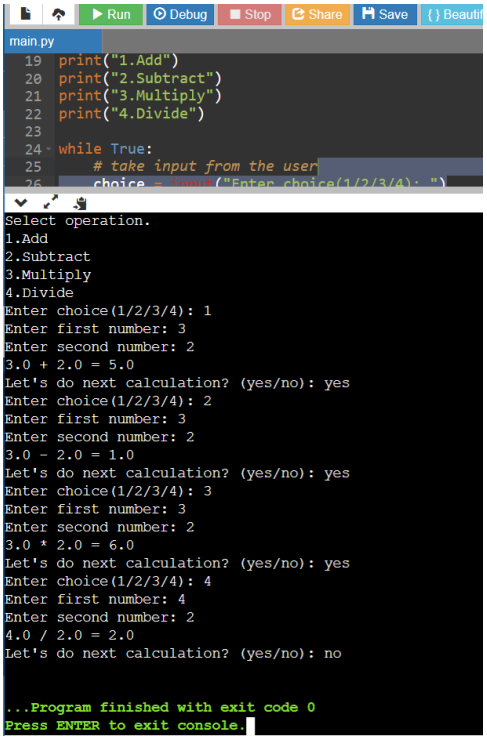
Enter choice(1/2/3/4): 4

Enter first number: 4

Enter second number: 2

4.0 / 2.0 = 2.0

Let's do next calculation? (yes/no): no



3. Write a program to concatenate, reverse and slice a string.

Solution:

```
# Defining strings
var1 = "Hello "
var2 = "World"

# + Operator is used to combine strings
var3 = var1 + var2
print(var3)

txt = "Hello World"[::-1] #Reverse of a string
print(txt)

b = "Hello, World!"
```

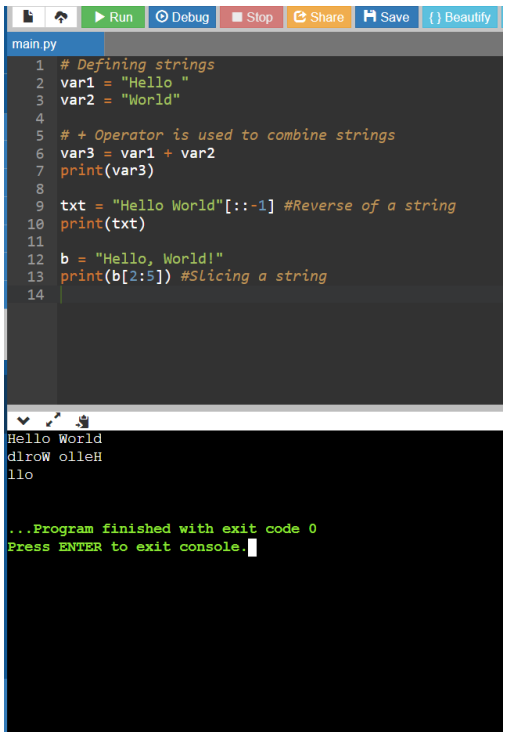
print(b[2:5]) #Slicing a string

Output:

Hello World

dlroW olleH

llo



4. Why is a python a popular programming language?

Solution:

- Python is an interpreted, object-oriented, high-level programming language with dynamic semantics.
- Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

- Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.
- Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

5. What are the other frameworks that can be used with Python?

Solution:

- Django
- Flask
- Dash
- Falcon
- CubicWeb
- CherryPy

6. Full Form of WSGI?

Solution:

- WSGI Stands for Web Server Gateway Interface
- WSGI is a specification that describes the communication between web servers and Python web applications or frameworks.
- It explains how a web server communicates with python web applications/frameworks and how web applications/frameworks can be chained for processing a request.